■ Reverse Number – Short Notes

■ Problem Statement

Given an integer **num**, reverse its digits. Leading zeros should be ignored in the output.

■ Examples

Example 1:
Input: num = 12345
Output: 54321

Example 2:
Input: num = 100
Output: 1 (because 001 is ignored)

Example 3:
Input: num = 78910
Output: 1987

Example 4:
Input: num = 123
Output: 321

■ Core Logic

1. Extract the last digit using **num % 10**.
2. Add it to reversed value using **ans = ans * 10 + digit**.
3. Remove last digit by **num = num / 10**.
4. Continue until num becomes 0.

■ Dry Run (num = 123)

Step	num	digit	ans
1	123	3	3
2	12	2	32
3	1	1	321

■ C++ Code

```
class Solution {
  public:
  int reverseNumber(int num) {
    int ans = 0;
    while (num != 0) {
      int digit = num % 10;
      ans = ans * 10 + digit;
      num = num / 10;
    }
    return ans;
  }
};
```

■■ Time & Space Complexity

Time Complexity: O(d) — d = number of digits
Space Complexity: O(1)

■ Key Notes

- Works for all positive integers.
- Leading zeros are automatically removed.
- Very common interview question.